Learning objectives, knowledge and skills

Advent		Lent		Pentecost	
3.1 Computing systems and networks – Connecting computers Identifying that digital devices have inputs, processes, and outputs, and how devices can be connected to make networks.	**3.2 Creating media – Desktop publishing** Creating documents by modifying text, images, and page layouts for a specified purpose.	**3.3 Programming A - Sequencing sounds** Creating sequences in a block-based programming language to make music	**3.4 Data and information – Branching databases** Building and using branching databases to group objects using yes/no questions.	**3.5 Creating media - Stop-frame animation** Capturing and editing digital still images to produce a stop-frame animation that tells a story.	**3.6 Programming B - Events and actions in programs** Writing algorithms and programs that use a range of events to trigger sequences of actions.

1. To explain how digital devices function • I can explain that digital devices accept inputs • I can explain that digital devices produce outputs • I can follow a process **2. To identify input and output devices** • I can classify input and output devices • I can describe a simple process • I can design a digital device **3. To recognise how digital devices can change the way we work** • I can explain how I use digital devices for different activities • I can recognise similarities between using digital devices and non-digital tools • I can suggest differences between using digital devices and non-digital tools **4. To explain how a computer network can be used to share information** • I can discuss why we need a network switch • I can explain how messages are passed through multiple connections • I can recognise different connections **5. To explore how digital devices can be connected** • I can demonstrate how information can be passed between devices • I can explain the role of a switch, server, and wireless access point in a network • I can recognise that a computer network is made up of a number of devices **6. To recognise the physical components of a network** • I can identify how devices in a network are connected together • I can identify networked devices around me • I can identify the benefits of computer networks	**1. To recognise how text and images convey information** • I can explain the difference between text and images • I can identify the advantages and disadvantages of using text and images • I can recognise that text and images can communicate messages clearly **2. To recognise that text and layout can be edited** • I can change font style, size, and colours for a given purpose • I can edit text • I can explain that text can be changed to communicate more clearly **3. To choose appropriate page settings** • I can create a template for a particular purpose • I can define the term 'page orientation' • I can recognise placeholders and say why they are important **4. To add content to a desktop publishing publication** • I can choose the best locations for my content • I can make changes to content after I've added it • I can paste text and images to create a magazine cover **5. To consider how different layouts can suit different purposes** • I can choose a suitable layout for a given purpose • I can identify different layouts • I can match a layout to a purpose **6. To consider the benefits of desktop publishing** • I can compare work made on desktop publishing to work created by hand • I can identify the uses of desktop publishing in the real world • I can say why desktop publishing might be helpful	**1. To explore a new programming environment** • I can explain that objects in Scratch have attributes (linked to) • I can identify the objects in a Scratch project (sprites, backdrops) • I can recognise that commands in Scratch are represented as blocks **2. To identify that commands have an outcome** • I can choose a word which describes an on-screen action for my plan • I can create a program following a design • I can identify that each sprite is controlled by the commands I choose **3. To explain that a program has a start** • I can create a sequence of connected commands • I can explain that the objects in my project will respond exactly to the code • I can start a program in different ways **4. To recognise that a sequence of commands can have an order** • I can combine sound commands • I can explain what a sequence is • I can order notes into a sequence **5. To change the appearance of my project** • I can build a sequence of commands • I can decide the actions for each sprite in a program • I can make design choices for my artwork **6. To create a project from a task description** • I can identify and name the objects I will need for a project • I can implement my algorithm as code • I can relate a task description to a design	**1. To create questions with yes/no answers** • I can create two groups of objects separated by one attribute • I can investigate questions with yes/no answers • I can make up a yes/no question about a collection of objects **2. To identify the attributes needed to collect data about an object** • I can arrange objects into a tree structure • I can create a group of objects within an existing group • I can select an attribute to separate objects into groups **3. To create a branching database** • I can group objects using my own yes/no questions • I can select objects to arrange in a branching database • I can test my branching database to see if it works **4. To explain why it is helpful for a database to be well structured** • I can compare two branching database structures • I can create yes/no questions using given attributes • I can explain that questions need to be ordered carefully to split objects into similarly sized groups **5. To plan the structure of a branching database** • I can create a physical version of a branching database • I can create questions that will enable objects to be uniquely identified • I can independently create questions to use in a branching database **6. To independently create an identification tool** • I can create a branching database that reflects my plan • I can suggest real-world uses for branching databases • I can work with a partner to test my identification tool	**1. To explain that animation is a sequence of drawings or photographs** • I can create an effective flip book—style animation • I can draw a sequence of pictures • I can explain how an animation/flip book works **2. To relate animated movement with a sequence of images** • I can create an effective stop-frame animation • I can explain why little changes are needed for each frame • I can predict what an animation will look like **3. To plan an animation** • I can break down a story into settings, characters and events • I can create a storyboard • I can describe an animation that is achievable on screen **4. To identify the need to work consistently and carefully** • I can evaluate the quality of my animation • I can review a sequence of frames to check my work • I can use onion skinning to help me make small changes between frames **5. To review and improve an animation** • I can evaluate another learner's animation • I can explain ways to make my animation better • I can improve my animation based on feedback **6. To evaluate the impact of adding other media to an animation** • I can add other media to my animation • I can evaluate my final film • I can explain why I added other media to my animation	**1. To explain how a sprite moves in an existing project** • I can choose which keys to use for actions and explain my choices • I can explain the relationship between an event and an action • I can identify a way to improve a program **2. To create a program to move a sprite in four directions** • I can choose a character for my project • I can choose a suitable size for a character in a maze • I can program movement **3. To adapt a program to a new context** • I can choose blocks to set up my program • I can consider the real world when making design choices • I can use a programming extension **4. To develop my program by adding features** • I can build more sequences of commands to make my design work • I can choose suitable keys to turn on additional features • I can identify additional features (from a given set of blocks) **5. To identify and fix bugs in a program** • I can match a piece of code to an outcome • I can modify a program using a design • I can test a program against a given design **6. To design and create a maze-based challenge** • I can evaluate my project • I can implement my design • I can make design choices and justify them

#BTK and Links with other subjects

Key Vocabulary